IN THE CLAIMS

Please amend the claims as follows:

Claim 1 (Currently Amended): A computer implemented log comparison debug support system which inputs a plurality of logs in which a series of events that have occurred as a result of the execution of a target program are recorded, and supports debugging by performing log comparison, the system comprising:

a condition specifying device configured to specify a condition for log comparison, which includes begin and end events in a partial log, and to specify an extraction rule for extracting at least part of the event sequence sandwiched between the begin and end events;

a partial log creating device configured to create a plurality of partial logs from the input logs according to the condition for log comparison and the extraction rule;

a master log creating device configured to create a master log by concatenating the partial logs;

a normalized log creating device configured to create normalized logs by normalizing said partial logs by use of the master log serving as a normalization reference;

a feature value computing device configured to compute feature values representing the degree of feature of the occurrence and nonoccurrence of said events for each of the normalized logs created by said normalized log creating device; and

a similarity computing device configured to compute, in a combination of a specific partial log and another partial log, the similarity between these partial logs by performing a specific operation based on said feature values.

Claim 2 (Canceled).

Claim 3 (Original): The system of claim 1, wherein said feature value computing device computes one feature value in one normalized log by referring to the other normalized logs.

Claim 4 (Original): The system of claim 1, wherein said similarity computing device computes the similarity between said combination of partial logs by inner product operation of respective feature values of the normalized logs.

Claim 5 (Original): The system of claim 1, wherein said feature value computing device computes probability of occurrence or nonoccurrence of events as said feature value.

Claim 6 (Previously Presented): The system of claim 5, wherein said similarity computing device computes the similarity between said combination of partial logs by adding or subtracting of the absolute value of a logarithm of respective feature values of the normalized logs.

Claim 7 (Original): The system of clam 1, further comprising:

a specifying device configured to specify one of said partial logs;

a selecting device, coupled between said similarity computing device and the specifying device, configured to select another partial log which is similar to said specified log according to the similarity computed by said similarity computing device.

Claim 8 (Currently Amended): A <u>computer implemented</u> log comparison debug support system which inputs a plurality of operation logs in which a series of events that have

occurred as a result of the execution of a target program are recorded, and supports debugging by performing log comparison, the system comprising:

a condition specifying device configured to specify a condition for log comparison, which includes begin and end events in a partial log, and to specify an extraction rule for extracting at least part of the event sequence sandwiched between the begin and end events;

a partial log creating device configured to create a plurality of partial logs from the input operation logs according to the condition for log comparison and the extraction rule;

a master log creating device configured to input a source program of said target program to create a master log by expanding the source program;

a normalized log creating device configured to create normalized logs by normalizing said partial logs by use of the master log serving as a normalization reference;

a feature value computing device configured to compute feature values representing the degree of feature of the occurrence and nonoccurrence of said events for each of the normalized logs created by said normalized log creating device; and

a similarity computing device configured to compute, in a combination of a specific partial log and another partial log, the similarity between these partial logs by performing a specific operation based on said feature values.

Claim 9 (Original): The system of claim 8, wherein said master log creating device comprises an expander configured to expand a description corresponding to a specific function in the source program, and create the result of the expansion as said master log.

Claim 10 (Original): The system of claim 9, wherein said expander expands a description of specific syntax including function calls and loops in said specific function.

Claim 11 (Canceled).

Claim 12 (Original): The system of claim 8, wherein said feature value computing device computes one feature value in one normalized log by referring to the other normalized logs.

Claim 13 (Original): The system of claim 8, wherein said similarity computing device computes the similarity between said combination of partial logs by inner product operation of respective feature values of the normalized logs.

Claim 14 (Original): The system of claim 8, wherein said feature value computing device computes probability of occurrence or nonoccurrence of events as said feature value.

Claim 15 (Previously Presented): The system of claim 14, wherein said similarity computing device computes the similarity between said combination of partial logs by adding or subtracting of the absolute value of a logarithm of respective feature values of the normalized logs.

Claim 16 (Original): The system of clam 8, further comprising:

a specifying device configured to specify one of said partial logs;

a selecting device, coupled between said similarity computing device and the specifying device, configured to select another partial log which is similar to said specified log according to the similarity computed by said similarity computing device.

Claim 17 (Currently Amended): A computer implemented method for supporting log comparison debugging, the method comprising:

inputting a plurality of logs in which a series of events that have occurred as a result of the execution of a target program are recorded;

specifying a condition for log comparison, which includes begin and end events in a partial log;

specifying an extraction rule for extracting at least part of the event sequence sandwiched between the begin and end events;

creating a plurality of partial logs from the input logs according to the condition for log comparison and the extraction rule;

creating a master log by concatenating the partial logs;

creating normalized logs by normalizing said partial logs by use of the master log serving as a normalization reference;

computing feature values representing the degree of feature of the occurrence and nonoccurrence of said events for each of the normalized logs; and

computing, in a combination of a specific partial log and another partial log, the similarity between these partial logs by performing a specific operation based on said feature values.

Claim 18 (Currently Amended): A computer implemented method for supporting log comparison debugging, the method comprising:

inputting a plurality of operation logs in which a series of events that have occurred as a result of the execution of a target program are recorded;

specifying a condition for log comparison, which includes begin and end events in a partial log;

specifying an extraction rule for extracting at least part of the event sequence sandwiched between the begin and end events;

creating a plurality of partial logs from the input operation logs according to the condition for log comparison and the extraction rule;

inputting a source program of said target program and creating a master log based on the source program by expanding the source program;

creating normalized logs by normalizing said partial logs by use of the master log serving as a normalization reference;

computing feature values representing the degree of feature of the occurrence and nonoccurrence of said events for each of the normalized logs; and

computing, in a combination of a specific partial log and another partial log, the similarity between these partial logs by performing a specific operation based on said feature values.

Claim 19 (Previously Presented): A computer program product configured to store program instructions for execution on a computer system enabling the system to perform:

inputting a plurality of logs in which a series of events that have occurred as a result of the execution of a target program are recorded;

specifying a condition for log comparison, which includes begin and end events in a partial log;

specifying an extraction rule for extracting at least part of the event sequence sandwiched between the begin and end events;

creating a plurality of partial logs from the input logs according to the condition for log comparison and the extraction rule;

creating a master log by concatenating the partial logs;

creating normalized logs by normalizing said partial logs by use of the master log serving as a normalization reference;

computing feature values representing the degree of feature of the occurrence and nonoccurrence of said events for each of the normalized logs; and

computing, in a combination of a specific partial log and another partial log, the similarity between these partial logs by performing a specific operation based on said feature values.

Claim 20 (Previously Presented): A computer program product configured to store program instructions for execution on a computer system enabling the system to perform:

inputting a plurality of operation logs in which a series of events that have occurred as a result of the execution of a target program are recorded;

specifying a condition for log comparison, which includes begin and end events in a partial log;

specifying an extraction rule for extracting at least part of the event sequence sandwiched between the begin and end events;

creating a plurality of partial logs from the input operation logs according to the condition for log comparison and the extraction rule;

inputting a source program of said target program and creating a master log based on the source program by expanding the source program;

creating normalized logs by normalizing said partial logs by use of the master log serving as a normalization reference;

computing feature values representing the degree of feature of the occurrence and nonoccurrence of said events for each of the normalized logs; and

computing, in a combination of a specific partial log and another partial log, the similarity between these partial logs by performing a specific operation based on said feature values.